ABSTRACT OF THE DISCLOSURE

A mechanism for compiling objects into representations of the objects, and for
decompiling the representations of the objects into copies of the objects, is described. A
virtual machine (e.g. the Java Virtual Machine (JVM)) may include extensions for
compiling objects, (e.g. Java Objects) into data representation language (e.g. XML)
representations of the objects, and for decompiling representations of objects into objects.
The virtual machine may supply an Applications Programming Interface (API) to the
compilation/decompilation extensions. The client and service may be executing within
virtual machines. The virtual machines may be on the same device or on different
devices. The compiler/decompiler API may accept an object as input, and output a data
representation language representation of the object and all its referenced objects (the
object graph) in a data stream. In addition, the compiler/decompiler API may accept a
data stream, which includes a representation of the object and all its referenced objects
(the object graph), and output the object (and all the objects in its object graph). In one
embodiment, an intermediary format may be used to represent a data representation
language document and may be dynamically processed to generate a class instance from
the data representation language document.